DOCUMENTATION

Abstract

The Student Grade Sheet Management System is developed to simplify and automate the process of managing student academic records. The system efficiently stores student details, subject marks, and grades, ensuring accurate calculation of totals, percentages, and results. By replacing the traditional manual method, it reduces paperwork, saves time, and minimizes human errors. The system provides quick access to student performance data and helps teachers and administrators manage results effectively. It is a reliable, user-friendly solution that improves accuracy and transparency in academic evaluation.

Introduction

In educational institutions, managing student grades manually is a difficult and time-consuming process. Errors in calculation and record maintenance can affect student performance evaluation. The Student Grade Sheet Management System replaces the traditional manual system with a digital platform that stores student information, subject marks, grades, and results securely. This system helps teachers and administrators save time and improve efficiency.

Software Requirements

Operating System: Windows / Linux

Programming Language: C / Java / Python

Database: MySQL / SQLite

IDE/Compiler: Turbo C / VS Code / Eclipse

Web Browser (if web-based): Google Chrome / Microsoft Edge

Hardware Requirements

Processor: Intel i3 or higher

RAM: Minimum 4 GB

Hard Disk: 250 GB or above

Input Devices: Keyboard, Mouse

Output Device: Monitor

Code

```
#include <stdio.h>


struct student {
   int roll;
   char name[30];
   int m1, m2, m3;
   int total;
   float percent;
   char grade;
   char result[5];
};
```

```
char gradeCalc(float p) {
    if (p >= 90) return 'A';
    else if (p >= 75) return 'B';
    else if (p >= 60) return 'C';
    else if (p >= 40) return 'D';
    else return 'F';
}

void sortByRoll(struct student s[], int n) {
    struct student temp;
    for (int i = 0; i < n - 1; i++) {
        for (int j = i + 1; j < n; j++) {
            if (s[i].roll > s[j].roll) {
                temp = s[i];
                s[i] = s[j];
                s[j] = temp;
            }
        }
    }
}

int main() {
```

```
struct student s[50];
int n;

printf("Enter number of students: ");
scanf("%d", &n);

for (int i = 0; i < n; i++) {
    printf("\nStudent %d Details\n", i + 1);

    printf("Roll No: ");
    scanf("%d", &s[i].roll);

    printf("Name: ");
    scanf("%s", s[i].name);

    printf("Marks (3 subjects): ");
    scanf("%d %d %d", &s[i].m1, &s[i].m2, &s[i].m3);

    s[i].total = s[i].m1 + s[i].m2 + s[i].m3;
    s[i].percent = s[i].total / 3.0;
    s[i].grade = gradeCalc(s[i].percent);

    if (s[i].grade == 'F')
```

```
            sprintf(s[i].result, "FAIL");
        else
            sprintf(s[i].result, "PASS");
    }

    sortByRoll(s, n);

    printf("\n-------------------------------------------------------------\n");
    printf("Roll  Name     Total  Percent   Grade   Result\n");
    printf("-------------------------------------------------------------\n");

    for (int i = 0; i < n; i++) {
        printf("%-5d %-9s %-6d %-8.2f%% %-7c %s\n",
               s[i].roll, s[i].name, s[i].total,
               s[i].percent, s[i].grade, s[i].result);
    }

    printf("-------------------------------------------------------------\n");

    return 0;
}
```

Result

```
Enter number of students: 4

Student 1 Details
Roll No: 1
Name: anagha
Marks (3 subjects): 98
93
95

Student 2 Details
Roll No: 2
Name: prasanna
Marks (3 subjects): 96
94
97

Student 3 Details
Roll No: 3
Name: neeraj
Marks (3 subjects): 93
95
98

Student 4 Details
Roll No: 4
Name: sandeep
Marks (3 subjects): 92
94
90

-------------------------------------------------------
-------
Roll  Name      Total  Percent   Grade   Result
-------------------------------------------------------
-------
1     anagha    286    95.33   % A       PASS
2     prasanna  287    95.67   % A       PASS
3     neeraj    286    95.33   % A       PASS
4     sandeep   276    92.00   % A       PASS
-------------------------------------------------------
```

Conclusion

The Student Grade Sheet Management System provides an efficient and reliable way to manage student academic records. It reduces manual work, minimizes errors in grade calculation, and saves time for teachers and administrators. The system ensures accurate storage and quick access to student performance data. Overall, it improves efficiency, accuracy, and transparency in academic result management.